Strong Authentication - Adaptive Authentication

Introduction:

Adaptive authentication is a secure and flexible form of authentication. It enables adjusting the authentication factors based on the risk probability associated with the access request. This enables ensuring security without impacting usability at the time of authentication thereby providing enhanced user experience.

WSO2 Identity Server allows you to configure appropriate authentication factors via a script. The authentication script editor of the WSO2 Identity Server Management Console enables configuring the authentication script and supports a set of **predefined templates to** easily set up the authentication logic.

These are the adaptive authentication scenarios supported in WSO2 Identity Server which can be used to configure the adaptive authentication.

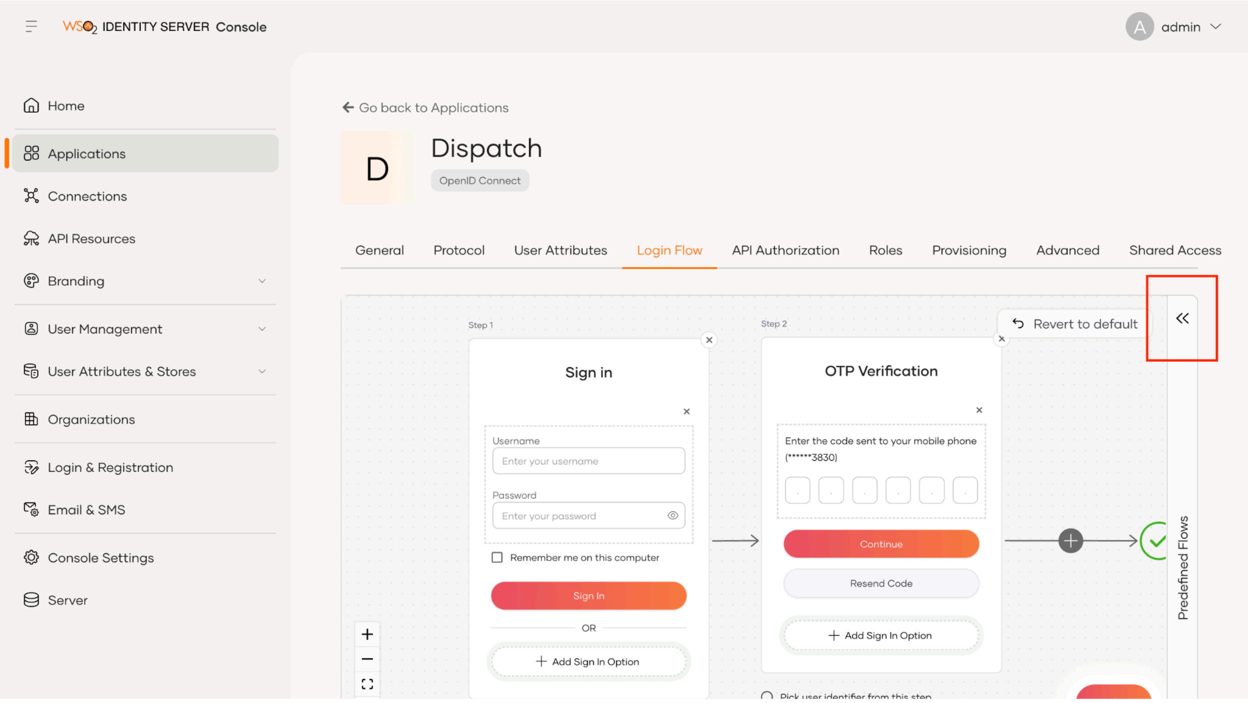
- User role based
- User store based
- Login attempts based
- User group based
- User device based
- IP address based

This tutorial guides you to configure the following adaptive authentication scenarios in WSO2 Identity Server using sample applications.

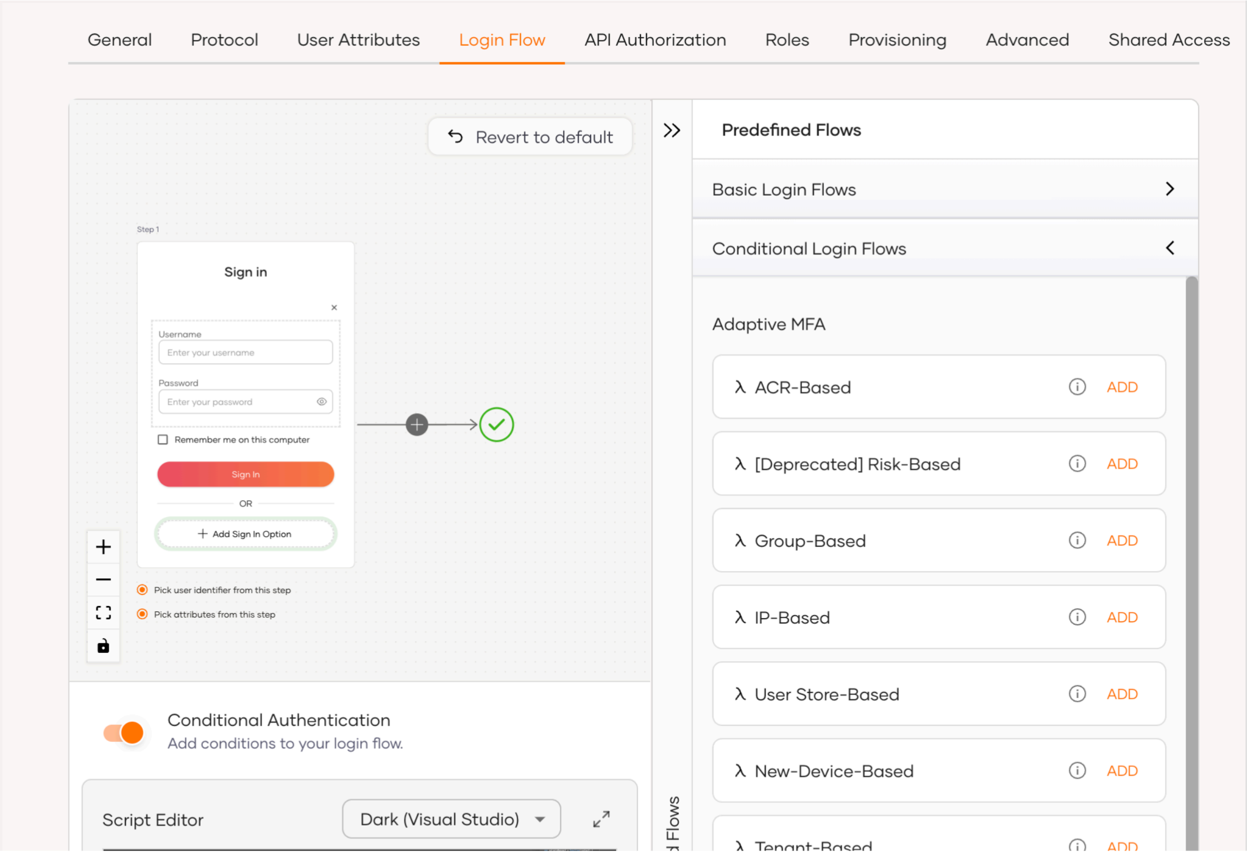
- User role based
- User device based
- IP address based

Configure Application with Adaptive Authentication script

1. In the WSO2 Identity Server **Console,** from the menu click **Applications.**
2. Click on the configured **Dispatch** Application**.**
3. Goto to the **Login Flow** tab.
4. Toggle the **Conditional Authentication** button.
   - For java 17 this button may not be visible as it is not enabled by default.
   - To enable Conditional Authentication refer this document https://is.docs.wso2.com/en/latest/deploy/enable-adaptive-authentication/.
5. Click on the **Predefined Flows.**



6. Collapse the **Basic Login Flow** and Expand **Conditional Login Flows**.



Conditional Authentication
Add conditions to your login flow.

User Role-Based Adaptive Authentication

Introduction:

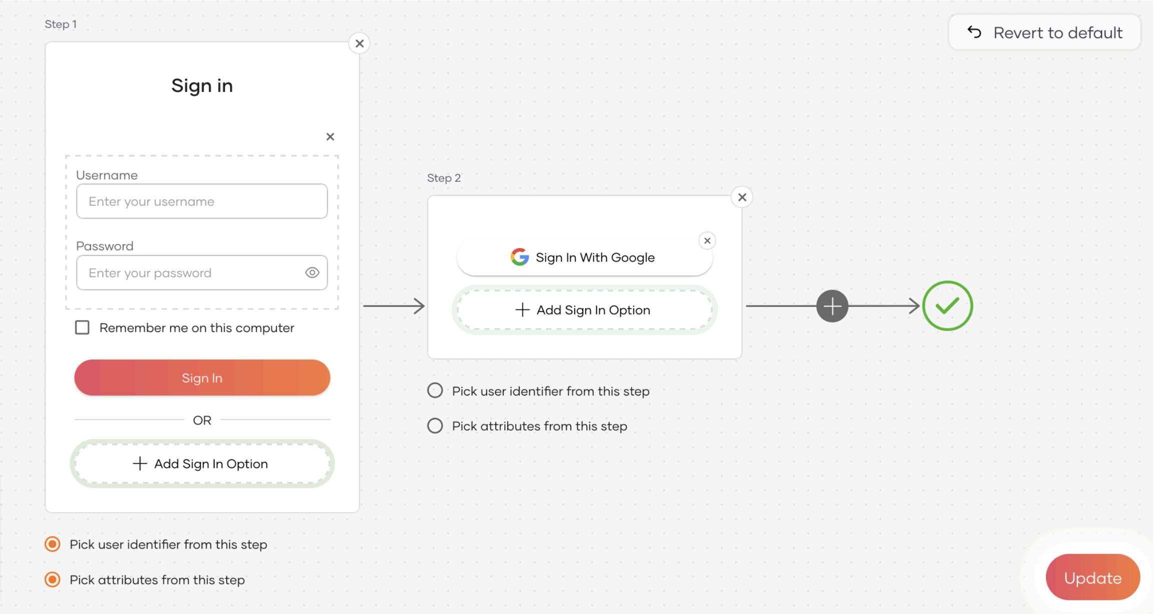
In this scenario we are going to create two users

1. Arya : Has `admin` and `manager` **application** roles
2. John : Has `director` **application** role

We will make the users who have the role `admin` or `manager` to do two factor authentication (2 FA).

Setting up:

1. Locate the **Authentication Script Editor** configured at Configure Application with Adaptive Authentication script section.
2. Then click **Role-Based -> ADD.**
3. As you can see, the allowed roles are added as **admin** and **manager** roles.
4. From the **Authentication Step Configuration**, first click on **Revert to default**, to clear and existing configuration.
5. After clearing add,
    a. **User and Password** as the first step.
    b. **Google** as the 2nd step.



Step 1:
- ● Pick user identifier from this step
- ● Pick attributes from this step

Step 2:
- ○ Pick user identifier from this step
- ○ Pick attributes from this step

6. Click **Update.**
7. Create the users **Arya** and **John** with proper roles as described in the **1. User, Group and Role Management tutorial**.

Try it:

1. Access http://localhost:8080/pickup-dispatch/
2. Login as **Arya.**
3. Both the configured steps will be displayed.
4. Login as **John.**
5. Only step 1 will be displayed.

User Device Based Adaptive Authentication

Introduction:

In this scenario, we are going to create a user

1. Snow: who can access Pickup Dispatch

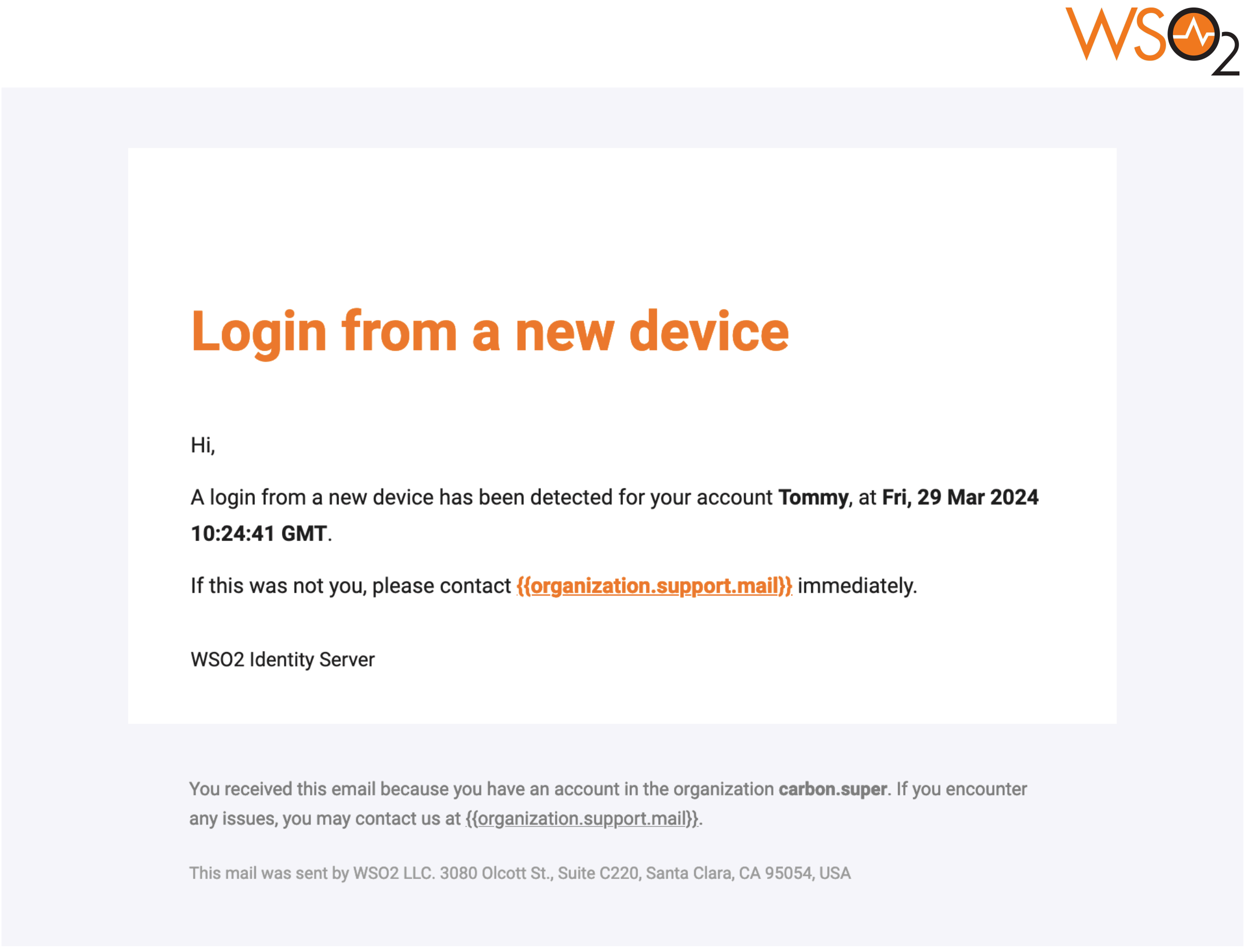
When Snow tries to login for the first time he will undergo 2FA and then the next time he can log in in a single step. This is done through cookies and their validation period is 2 years by default but can be changed. So when it expires again he has to undergo a 2FA.

Setting up:

1. Locate the **Authentication Script Editor** configured at Add Authentication script.
2. Then click **New-Device-Based -> ADD.**
3. From the **Authentication Step Configuration** configuring view add **Google** and remove all other authenticators.
4. Click **Update.**

Try it:

1. Access http://localhost:8080/pickup-dispatch/.
2. Login as **'Snow'.**
3. Both the configured steps will be displayed.
4. Logout and retry within the same browser.
5. Only step 1 will be displayed.
6. Now try with a different browser.
7. Both the configured steps will be displayed and once you login you will receive an email with the login timestamp.



Login from a new device

Hi,

A login from a new device has been detected for your account **Tommy**, at **Fri, 29 Mar 2024 10:24:41 GMT**.

If this was not you, please contact **{{organization.support.mail}}** immediately.

WSO2 Identity Server

You received this email because you have an account in the organization **carbon.super**. If you encounter any issues, you may contact us at {{organization.support.mail}}.

This mail was sent by WSO2 LLC. 3080 Olcott St., Suite C220, Santa Clara, CA 95054, USA

IP-Based Adaptive Authentication

Introduction:

In this scenario, we are going to create a user

1. Daenerys: who can access Pickup Dispatch

We will add our IP address into the recognized address range and login in one step. Then we will have a different address range which does not have our IP address. Now when we log in we have to go through 2FA.

Setting up:

1. Locate the **Authentication Script Editor** configured at Add Authentication script.
2. Then click **IP-Based -> ADD.**
3. Add this range '**127.0.0.0/16**' to the **var corpNetwork** (make sure your IP address range is captured here).
4. Click **Update**.
5. From the **Authentication Step Configuration** configuring view add **Google** and remove all other authenticators.
6. Click **Update.**
7. Now create the users Daenerys

Try it:

1. In the browser go to http://localhost.com:8080/saml2-web-app-dispatch.com
2. Now try to login using **Daenerys's credential**
3. You will be logged in one step

4. Now again follow the steps to configure IS but now don't add your range in **step 3**. (**remove the address range corresponding to you**)
5. Again try to login using **Daenerys's credential**
6. This time after basic login, you will be redirected to another login page
7. Enter the credentials to google login and click **Sign In**.
8. Now you have login using 2 steps.